

ユーザー事例から見る テスト自動化ツールの効果的な活用方法

2019年1月25日

株式会社アシスト システム基盤技術統括部

福地 義博

超サポ
愉快カンパニー

アシスト

アジェンダ

1. 会社紹介
 1. 株式会社アシスト
 2. アシストにおけるテストツール取扱いに関して
 3. 講師紹介
2. 事例紹介の前に
3. 製品紹介
 1. Unified Functional Testing
 2. LoadRunner
4. 活用事例紹介
 1. 事例 1 : 通信業様回帰テスト事例
 2. 事例 2 : 印刷業様負荷テスト事例

会社紹介

超サポ
愉快カンパニー

アシスト

会社概要－株式会社アシスト

設立	1972年3月
代表取締役会長	ビル・トッテン
代表取締役社長	大塚辰男
資本金	6,000万円
売上高	264億円（2017年度）
従業者数	1,120名（2018年4月現在） ※グループ会社含む
事業内容	コンピュータ用パッケージ・ソフトウェアの販売、技術サポート、教育 およびコンサルティング
本社所在地	東京都千代田区九段北4-2-1 市ヶ谷東急ビル
オフィス所在地	札幌、仙台、名古屋、金沢、大阪、広島、宇部、福岡、沖縄
取引会社数	6,100社（2017年度）
主要取扱製品数	60製品（2018年4月現在）
グループ会社	株式会社アシスト本舗 株式会社アシスト北海道 株式会社のれん

アシストとは

お客様に「ITソリューション」と「ソフトウェア・サポート」を提供する
「パッケージ・インテグレーター」です

データベース

- データベース
- Oracle DB運用管理
- Postgres開発支援
- データベース仮想化
- 企画／設計

クラウド

- クラウド基盤
- SaaS型運用管理

ITサービスマネジメント

- 品質強化対策
- システム運用管理

情報活用

- BI
- DWH
- データ連携
- データ・プレパレーション
- Webサイト管理（CMS）
- ビジネスルール管理（BRMS）
- 要求定義

アシスト 認定ソフトウェア

セキュリティ

- 情報漏洩対策
- インターネット分離
- ログ管理
- ログ分析
- ID管理

クライアント仮想化

- デスクトップ仮想化／アプリケーション仮想化

アシストにおけるテストツールの取り扱いについて

1997

負荷テストツール LoadRunner等の取り扱い開始

2002

機能テストツール「Quick Test Pro」国内リリース
※現 機能テスト・GUI処理自動化製品「Unified Functional Testing」

2019

メーカーよりテスト分野でアワード受賞（2017）
国内唯一のパートナー最高位「Platinum Partner」に認定

様々な
導入事例
活用事例

豊富な導入・構築
実績（累計）

UFT 約 700社
LR 約 300社

専任技術体制

- ・専任技術者配置
- ・専任サポートセンター
- ・プリセールス認定技術者
- ・テクニカル認定技術者

講師

氏名：福地 義博

趣味：海外旅行、写真、語学習得など

経歴：

2007年 株式会社アシスト入社

2008年 現場SE担当としてテスト案件に従事（現在に至る）

取得資格：・ JSTQB Foundation Level

案件実績：・ 回帰テスト（製造業、通信業、銀行業、保険業等）
・ 負荷テスト（航空運輸業、印刷業、電気・ガス・熱供給業等）





事例紹介の前に

超サポ
愉快カンパニー

アシスト

JSTQBに基づく開発とテストプロセス

計画とコントロール

分析と設計

実装と実行

終了基準の評価と レポート

テスト計画

- ・ スコープとリスクを特定し、テストの目的を定義
- ・ テストに対する包括的なアプローチを定義
- ・ テストの活動を定義
- ・ 定義した活動にリソースを割り当て
- ・ テストの分析と計画の活動をスケジュールリング

テストのコントロール

- ・ テスト結果の計測と分析
- ・ テストのモニタリングと文書化
- ・ 結果修正の管理
- ・ 意思決定

JSTQBに基づく開発とテストプロセス

計画とコントロール

分析と設計

実装と実行

終了基準の評価と
レポート

- ・テストベースをレビュー
- ・テストベースやテスト対象のテスト容易性を評価
- ・テスト分析に基づいて、テスト条件を識別し優先順位を付け
- ・高度なテストケースを設計し、優先順位を付け
- ・必要なテストデータを識別
- ・テスト環境を設計し、必要となるインフラストラクチャやツール類を識別
- ・テストベースとテストケースの間でトレーサビリティを確保

JSTQBに基づく開発とテストプロセス

計画とコントロール

分析と設計

実装と実行

終了基準の評価と
レポート

- ・ テストケースを決定し、実装し、優先順位を付け
- ・ テスト手順を開発し、優先順位を付け、テストデータを作成
- ・ 効率よくテストを行うため、テスト手順をベースにしてテストスイートを作る
- ・ テスト環境を正しくセットアップしたことを確認
- ・ テストベースとテストケースの間で双方向のトレーサビリティを確認し更新
- ・ 計画した順番に従い、テスト手順を人力、もしくはテストツールで実行
- ・ テストの実行結果の記録を取り、テスト対象のソフトウェア、テストツール、テストウェアのIDとバージョンを記録
- ・ 実行結果と期待する結果を比較

JSTQBに基づく開発とテストプロセス

計画とコントロール

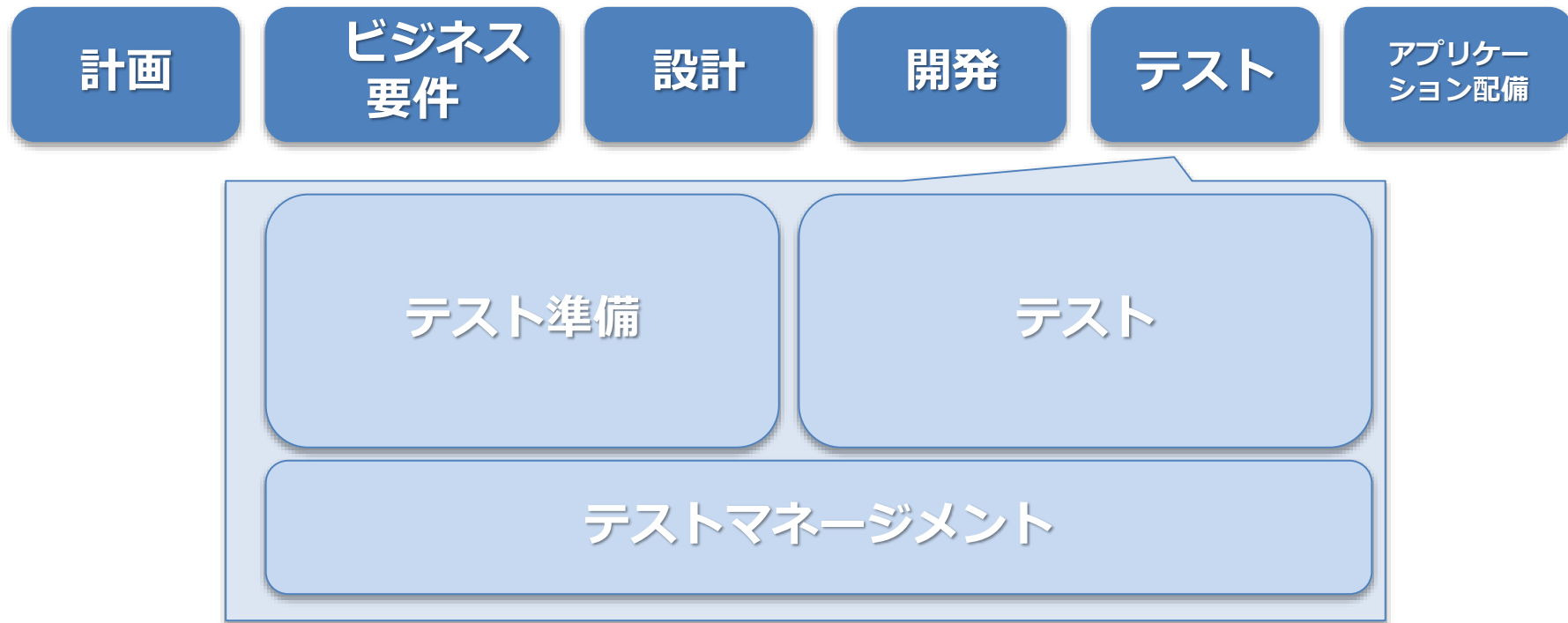
分析と設計

実装と実行

終了基準の評価と
レポート

- ・ 計画にあるテストの成果物がリリースされたかをチェック
- ・ システムを受け入れるために文書を作成
- ・ 次回も使えるように、テストウェア、テスト環境、テストのインフラストラクチャをまとめ、文書化
- ・ テストウェアを保守部門に引き渡し
- ・ 次回のリリースやプロジェクトのために教訓とすべきことをまとめる、その情報をテストの成熟度を改善するために利用

JSTQBに基づく開発とテストプロセス



JSTQBに基づく開発とテストプロセス



用語の整理

今回のセッションにおける用語を説明します

<用語解説>

回帰テスト（regression testing）：システムに変更作業を実施した場合に、以前まで正常に機能していた部分に不具合や影響が出ていないかを検証するテスト。

性能テスト（performance testing）：システムとしての要件が満たされているか検証するために、システム要件定義で定められた機能や能力がすべて備わっているかを確認するテスト。



製品紹介

超サポ
愉快カンパニー

アシスト

Unified Functional Testingとは

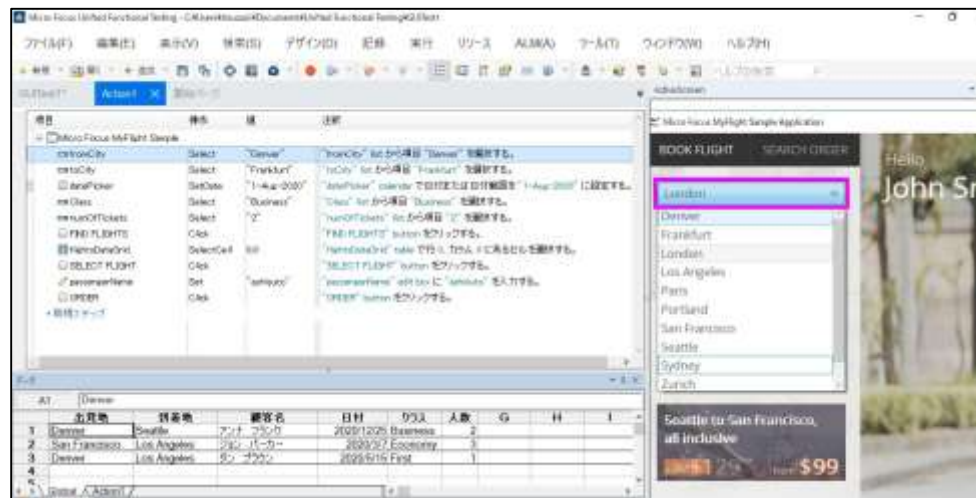


概要

- GUIアプリケーションの機能テストを効率化するキャプチャ&リプレイツール
- 回帰テスト等の繰り返しテストを自動化し、テスト実施とエビデンス取得の工数を削減
- 繰り返し行う操作を自動化することから定例業務の自動化でも利用されている

機能

- 操作時のエビデンスの取得も自動化
- 期待値に対するテスト実行結果を比較するチェックポイント機能を提供
- 画面への入力値は実行時に、Excelからデータを読み込み投入可



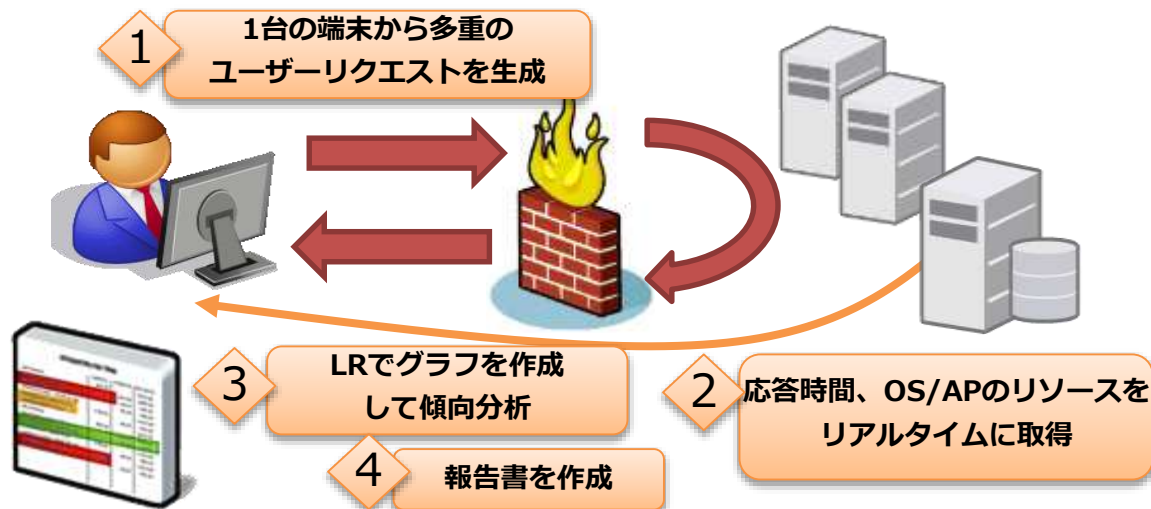
LoadRunnerとは

概要

- ・ 最小限の人、PC、時間のリソースで実環境に即した負荷テストを実現
- ・ 幅広い対応環境（Web、Citrix、SAP等）にて、負荷テスト標準ツールとして利用可能

機能

- ・ 入力値、タイミング等の実際の使用状況を緻密に再現可能、レポートも自動生成可能
- ・ 負荷テスト時に負荷対象サーバのリソース使用量をオンラインでモニタリング可能
- ・ 負荷テストレポートの自動生成機能により、多角的に分析が可能



事例 1 : 通信業様回帰テスト事例

通信系A社様の事例背景

■対象システム

通信系基幹システム（Web、VB、Javaなど）

■システム概要

以下の特徴を持つシステムがある

- ✓ 基幹系システムのため、高品質化が求められる
- ✓ 回帰テストの比率が高く、人海戦術でのテストに限界有り（テスト比率の約25%）
- ✓ テスト計画や環境面も都度準備が必要
- ✓ Windows10対応化も求められている



回帰テストにおける全体課題と対応検討策

- 回帰テストにおける全体課題と対応指針
お客様の課題を解決するために対応方針を検討

課題

① 手作業の課題

② テスト準備の
課題

検討策

① 回帰テストツールの導入

② テスト計画見直し、環境整備など
の方法論の検討

①手作業における課題と対応策

■手作業における課題と対応策

回帰テストツールUnified Functional Testing（UFT略）を導入

課題

手作業のため
対応に限界有

テストターの
属人化



対応策

回帰テストツールの導入により
作業品質の均一化と脱属人化

②テスト準備における課題

環境準備に おける課題



- ・ テストデータ
テストデータの作成にも時間がかかる
- ・ テスト環境
専用環境が用意できず複数の環境が
占有できない

テスト準備に おける課題



- ・ 回帰テスト範囲の抽出作業
都度テスト範囲を検討し、
観点・項目・シナリオを作成

②テスト準備における対策

環境準備に おける対策



- ・ テストデータ
UFTによるテストデータ作成自動化
- ・ テスト環境
回帰テスト用専用環境を構築し占有

テスト準備に おける対策



- ・ 回帰テスト範囲の抽出
回帰テストそのものを全部自動化
することとし、プロセス検討を廃止

回帰テストツール利用による効果

Before



After



ツールと方法論を利用/検討することで、
テスト実施のプロセスを全て自動化

UFTのテスト結果エビデンスの自動取得機能

処理内容一覧

The screenshot displays the HP Run Results Viewer interface. On the left, a tree view lists test steps, with 'toCity Select' highlighted. On the right, a table shows the results for the 'toCity Select' step, indicating it was successful. Below the table, a screenshot of the 'HP MyFlight Sample Application' is shown, displaying a flight booking interface with fields for origin (Denver), destination (Sydney), date (2017/06/22), class (Economy), and tickets (1). The application also shows a greeting 'Hello John Smith'.

オブジェクト	詳細	結果	時間
toCity Select	"Sydney"	完了	2017/06/21 - 14:47:37

結果詳細

実行時の
画面キャプチャ

専用Viewerでの参照の他に、
HTML、PDF、Word形式での
出力が可能。

実行結果ステータス(成功/失敗)と実行時のエビデンスを確認可能
※エラー発生時の画面も確認可能

回帰テストツールの技術課題

■技術課題

UFTを用いることで、回帰テストの自動化を目指したが以下の技術課題が発生

- ✓ UFTがアプリケーションを認識しない
- ✓ 既存テスト手順が自動化できない
- ✓ 認識できない画面や機能が発生

回帰テストツールの技術課題と対策

■技術課題と対策

UFTを利用時に発生した課題は以下の工夫を行うことで解決

アプリケーションを
UFTが認識しない



UFTとアプリケーションを
同一筐体に設置

テスト手順通りに
自動化ができない



実施手順を見直すこと
で自動化

一部認識できない
部品がある



VBSや画像認識での自動化

段階を踏んだテストツールの適用

フェーズ1

必要最低限のテスト（100件）

- ・ 最低限網羅しなければならない基本機能を中心としたテストを自動化
- ・ テストデータ作成用のシナリオもテストツールにて自動化

フェーズ2

回帰テスト（1000件）

- ・ 回帰テスト用の試験環境を構築し、テストデータ含めて環境面も配慮
- ・ 環境面の配慮により、効率化された作業時間にて回帰テストの網羅率を向上

フェーズ3

全回帰テスト（3000件）

- ・ テスト観点、項目、シナリオ作成のテスト準備作業を廃止し、回帰テスト自体を全部自動化

テストツール導入による効果のまとめ

工数削減率	テストカバレッジ	その他
20%削減 ※全テスト範囲	100%削減 ※回帰テスト範囲	・テスト結果エビデンス取得自動化 ・Windows10アップデートのテスト自動化

**テスト工数の削減および
テストカバレッジの向上も実現**

回帰テスト事例のまとめ

回帰テストのプロセス自体を全て自動化することで、テスト工数の削減およびテストカバレッジの向上を実現し、システム品質向上にも貢献

ポイント

テスト要件の発想転換により、準備工数の削減も実現

テスト自動化だけでなく
テストデータ作成やエビデンス取得等も自動化

方法論とテストツール導入により回帰テスト
自動化の全網羅を実現

参考：回帰テストを円滑に進めるポイント（1/2）

■事前の動作検証

- ✓ 事前の動作確認を行いツールの適用範囲を見定める

■既存テスト手順の見直し

- ✓ 既存テスト手順を見直し自動化に向けた最適なテスト設計を行う

■製品の習得コスト

- ✓ テストツールを習得する時間（コスト）を意識する
- ✓ 特に初回は習得までのスキルや開発コストが発生するため、習得コストも算出する

■適用範囲

- ✓ まずはパイロットプロジェクトで小さく始め、見極めを行う

参考：回帰テストを円滑に進めるポイント（2/2）

■期待値の明確化

- ✓ 期待値が明確でないテストケースは明確化する

■UI変更に伴う既存自動化資産のメンテナンス

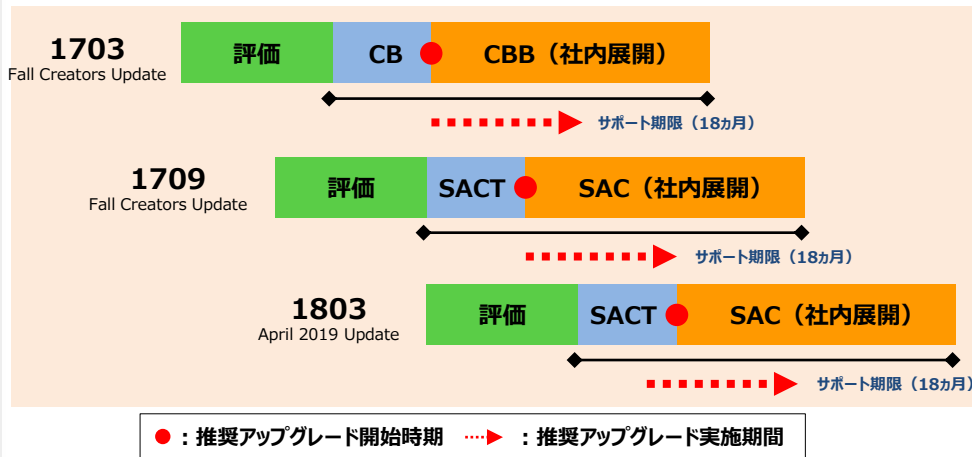
- ✓ UFTはオブジェクト認識型のテストツールであるが、UI変更時は修正が必要な場合があり、該当の属性値に影響がないか確認する
- ✓ テストケースとテストスクリプトが、1対1の関係だと、メンテナンスが大変になる可能性があり、部品化を行う

■テスト推進者/チームの設置

- ✓ 自動化を推進する人やチームを設置する
- ✓ パイロットプロジェクトで得た自動化の知識をドキュメント化する

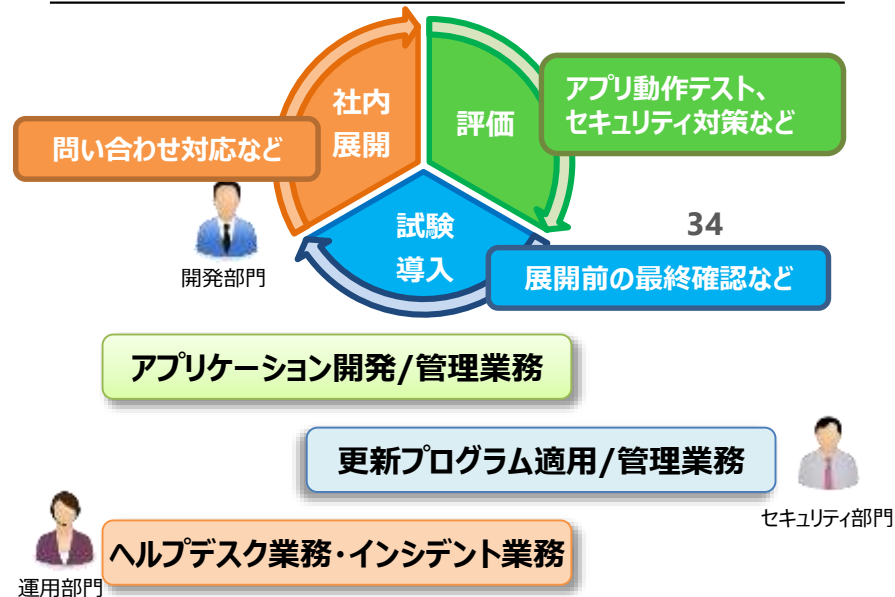
参考：Windows10におけるアップデートサイクル

WaaSによる定期的・継続的なアップデートサイクル



※今後は年2回（3月と9月）更新モデルがリリースされることになります。
※上図：マイクロソフト社の公開情報を基にアシストにて作成

アップデートサイクルで必要となる業務プロセス



年2回の大型アップデートの動作検証でも適用可能

参考：Windows10の適用事例

項目	内容
ツール利用頻度	大きく4回実施 ・マイグレーション時（VB化） ・機能改修時 ・Windows10対応 ・Windows10 1709対応
対象システム（環境）	・在庫管理システム（.NET） ・ワークフローシステム（Web）
対象機能	主に以下の11機能を自動化した ・データ投入系 ・画面制御系
自動化に対する工夫 1	自動化に対する戦略を検討し、テスト仕様書に伴う最低限網羅しなければならないパターンを自動化した
自動化に対する工夫 2	・テストケースの可否判定は、ツールには最低限判定機能の項目のみを実装した ・システムを改修し、テスト結果を比較する機能を実装した

事例 2 : 印刷業様負荷テスト事例

印刷業A社様の事例背景

■対象システム

年賀状システム（外向けWebサイト）

年賀状システム（店舗向けC/Sサイト）

■システム概要

提供サービスとしては最大規模のシステムである

- ✓ 繁忙期には800人程度のアクセスが集中している
- ✓ スマートフォン向けのWebサイトも展開している
- ✓ 店舗向けのシステムは全国300店舗に展開している



印刷業A社様の課題

■課題

2015年にアプリケーションが30分間停止するトラブルが発生

✓ アプリ担当とインフラ担当にて意見が相違し、調査が難航



印刷業A社様の課題に対しての対応策と問題点

■対応策

- ✓ 負荷テストツールLoadRunnerの検討
- ✓ アプリケーションモニタリングツール（APMツール）JENNIFERの検討

■テスト結果

問題がアプリケーションかどうかの切り分けが重要なポイントとなっていたがアプリケーションに問題があることが明確となった

■問題点

負荷テストを実施した際はレスポンスやサーバリソースが悪いところまでは把握でき、その後の調査で以下の課題が鮮明になった

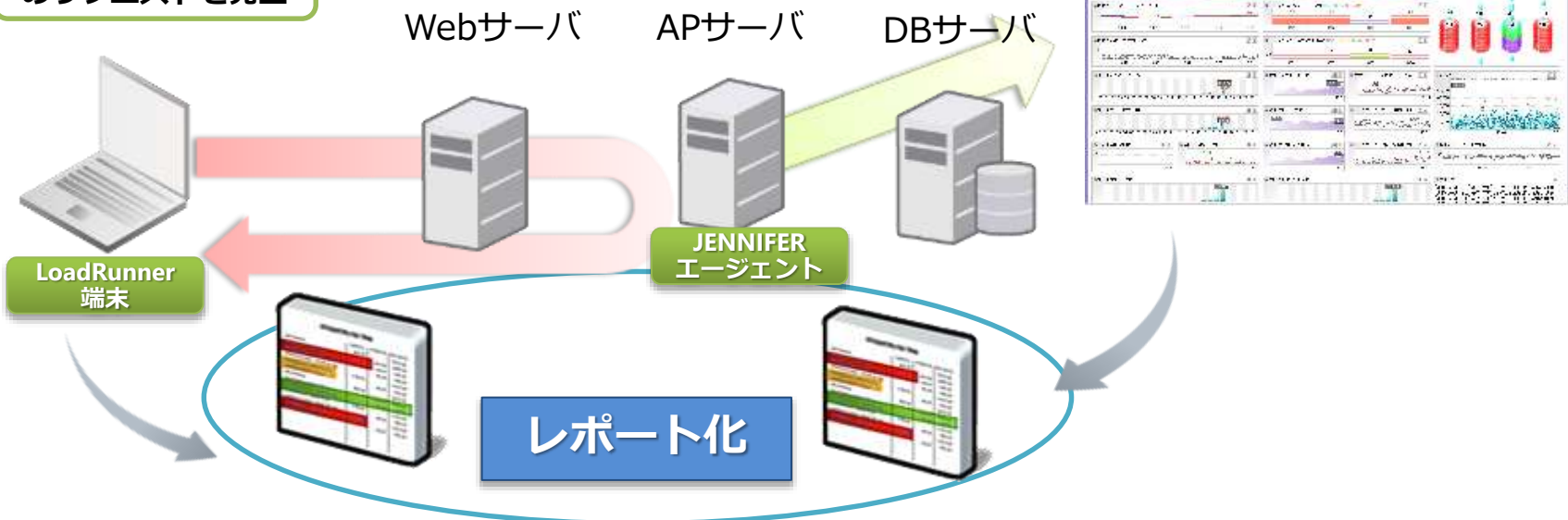
- ✓ Web、DBサーバのCPU高騰
- ✓ 特定のSQLのみが遅延
- ✓ 特定のテーブルに対する処理による応答時間の劣化

LoadRunner+JENNIFERシステム構成

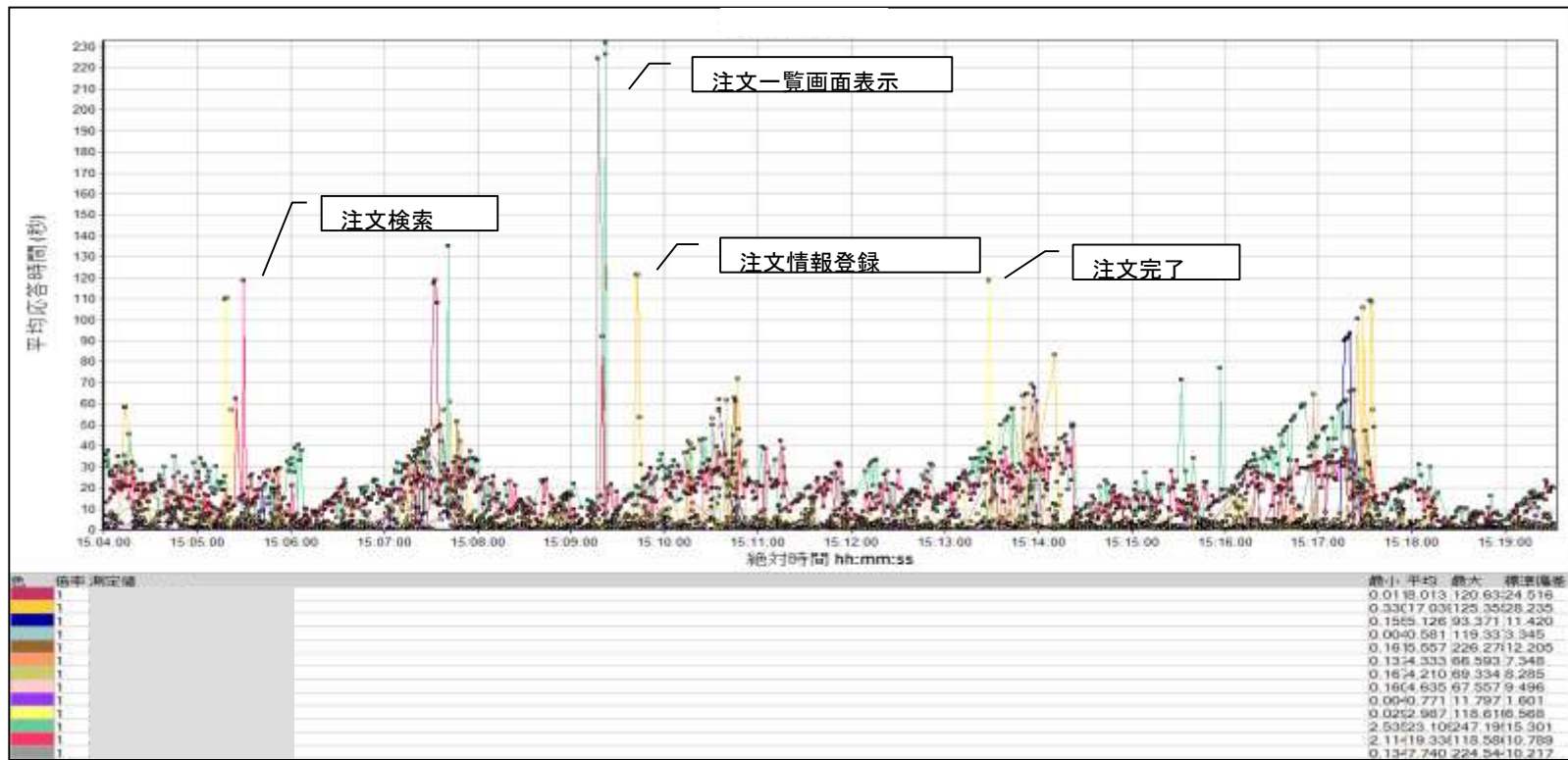
LoadRunnerで負荷テスト

JENNIFERで確認

1台の端末から多重
のリクエストを発生



LoadRunnerの応答時間グラフの例



印刷業A社様の対応策

JENNIFERにて以下の観点でエラー内容を詳細に調査

負荷テスト時間の応答時間分布

アプリケーション、SQL時間

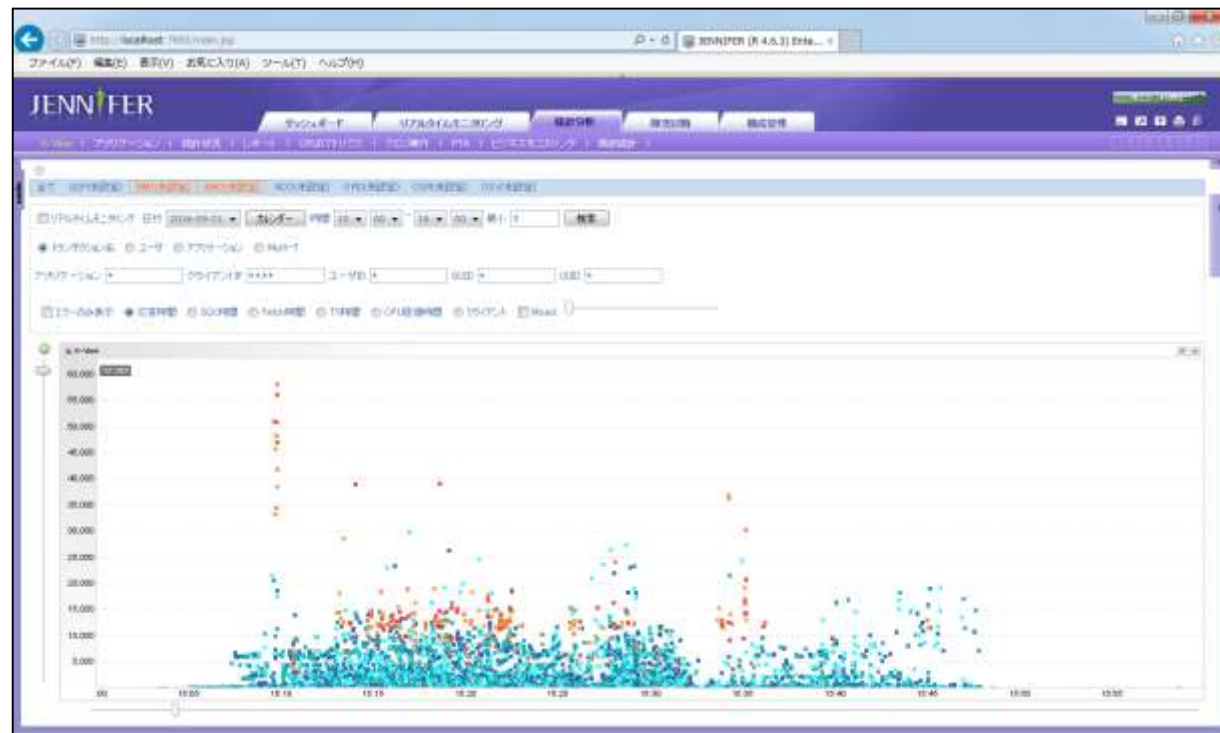
Exception情報

SQLException情報

アプリケーションのCRUD情報

JENNIFERの応答時間分布図の例

数百ユーザーでの実行時間[15:15~15:30]



JENNIFERによるログの解析結果例

■アプリケーション実行結果一覧8秒以上

アプリケーション名	最大応答時間
/XXXXXXXXXX/合成処理 1	58.141
/XXXXXXXXXX/注文内容確認処理	39.218
/XXXXXXXXX/合成処理 2	10.156

JENNIFERによるログの解析結果例

■SQL Exception

アプリケーション名	発生Exception	件数
/XXXXXXX/注文内容確認処理	ファイル参照	138
/XXXXXXXX/合成処理 1	ファイル参照	4

結果

アプリケーション開発元との調査を行い、以下の対応を実施

注文画面における
SELECT時のロックを
明示的に実施しない

優先度が高い処理のため、即改善対応し
効果測定となった

WEBサーバ上での
合成処理実施箇所を
キューイング化

- ・ 複数処理も同時並行して行っているため、高負荷時にCPUの負荷が上がっていた可能性が考えられる
- ・ 待ち行列化することにより、WEBサーバスケールアウトの効果を期待

負荷テスト事例のまとめ

LoadRunner単体だけでなく、JENNIFERも用いることで、ボトルネックの解析調査をより容易にでき、システム改修を行うことが出来ました

ポイント

負荷テストを行うことで、
事前に応答遅延に関する問題の切り分けが行えた

APMツールを用いることで、応答の悪い処理の問題特定をより詳細に把握可能し、チューニングが容易になった

参考：負荷テストを円滑に進めるポイント（1/2）

負荷テストを行う際によくある問題点に対してのポイントをお伝えします

項目	内容	検討項目
DBのデータ量	DBの蓄積データ量に伴い、処理性能が変動する	システムの想定蓄積データ量に応じて投入し、負荷テストで検証する
ユーザーアカウントと権限	ユーザーアカウントや権限が準備不足で負荷テスト時に権限に関するエラーが出る	リハーサル時において、稼働確認を行う。負荷テスト用のテストユーザーアカウントを準備し、ユーザーを増殖する
テストデータの準備	テストデータに不備があり、システムエラーが発生する	システムで利用可能なテストデータを事前に洗い出す
テストデータ量	負荷テスト時に利用するデータが不足し、負荷がかけられない	・ 負荷テストの実施回数や目標処理件数に応じて、データを準備する ・ テスト失敗時も考慮し、データを余分に準備する（例、20%増など）
サーバリソース	テスト実行後に取得すべきリソース値が取得できていない	リハーサル時や負荷テスト当日に確認し、必要なリソース値が取得できていることを確認する

参考：負荷テストを円滑に進めるポイント（2/2）

項目	内容	検討項目
負荷量の調整 （処理開始方法）	処理開始時の負荷量によっては、 処理が失敗する	開始時の処理量を緩やかに設定（ランプアップ）する
負荷量の調整 （ThinkTime）	想定負荷量に比べて、過負荷な状態や負荷が少ない事象が発生する	ThinkTimeの見直しを行い、調整を行う
ロードバランサー設定	ラウンドロビン等の設定を行っているが、向き先が偏ってしまう等の問題が発生する	・スクリプト内にCookie情報やサーバのアドレス等が含まれていないかを確認する ・IPアドレスから振り分けている場合は、LoadGeneratorの端末台数を増やし、均等になるように調整する
想定負荷がかからない （ネットワーク）	ネットワークI/O起因で、ネットワークがボトルネックになり、想定負荷がかからない	ネットワーク帯域の拡張および1ユーザーでのデバッグ時にトラフィック量を確認する
想定負荷がかからない （ハードウェア）	CPUやメモリが張り付き、想定負荷がかからない	負荷テストツールが推奨するハードウェア端末を予備台数含めて準備する

最後に

ご清聴頂きありがとうございました

本セッションに関するお問い合わせは以下の宛先までご連絡ください

株式会社アシスト 東日本技術本部 システム基盤技術統括部

E-mail : sk_info@ashisuto.co.jp

- 製品詳細（UFT）：

<https://www.ashisuto.co.jp/product/category/quality/functional-testing/>

- 製品詳細（LR）：

<https://www.ashisuto.co.jp/product/category/quality/loadrunner/>

※記載されている会社名、製品名は、各社の商標または登録商標です

Copyright© 2019 K.K. Ashisuto All Rights Reserved.

